Challenges in the Development and Evolution of Secure Open Architecture Command and Control Systems

Walt Scacchi and Thomas Alspaugh
Institute for Software Research
University of California, Irvine
Irvine, CA 92697-3455 USA

ISR Institute for Software Research
UNIVERSITY OF CALIFORNIA, IRVINE

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **JUN 2013**	2. REPORT TYPE	3. DATES COVERED **00-00-2013 to 00-00-2013**

4. TITLE AND SUBTITLE **Challenges in the Development and Evolution of Secure Open Architecture Command and Control Systems (BRIEFING CHARTS)**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of California, Irvine,Institute for Software Research,Irvine,CA,92697-3455**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

Presented at the 18th International Command & Control Research & Technology Symposium (ICCRTS) held 19-21 June, 2013 in Alexandria, VA. U.S. Government or Federal Rights License

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**25**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

Overview

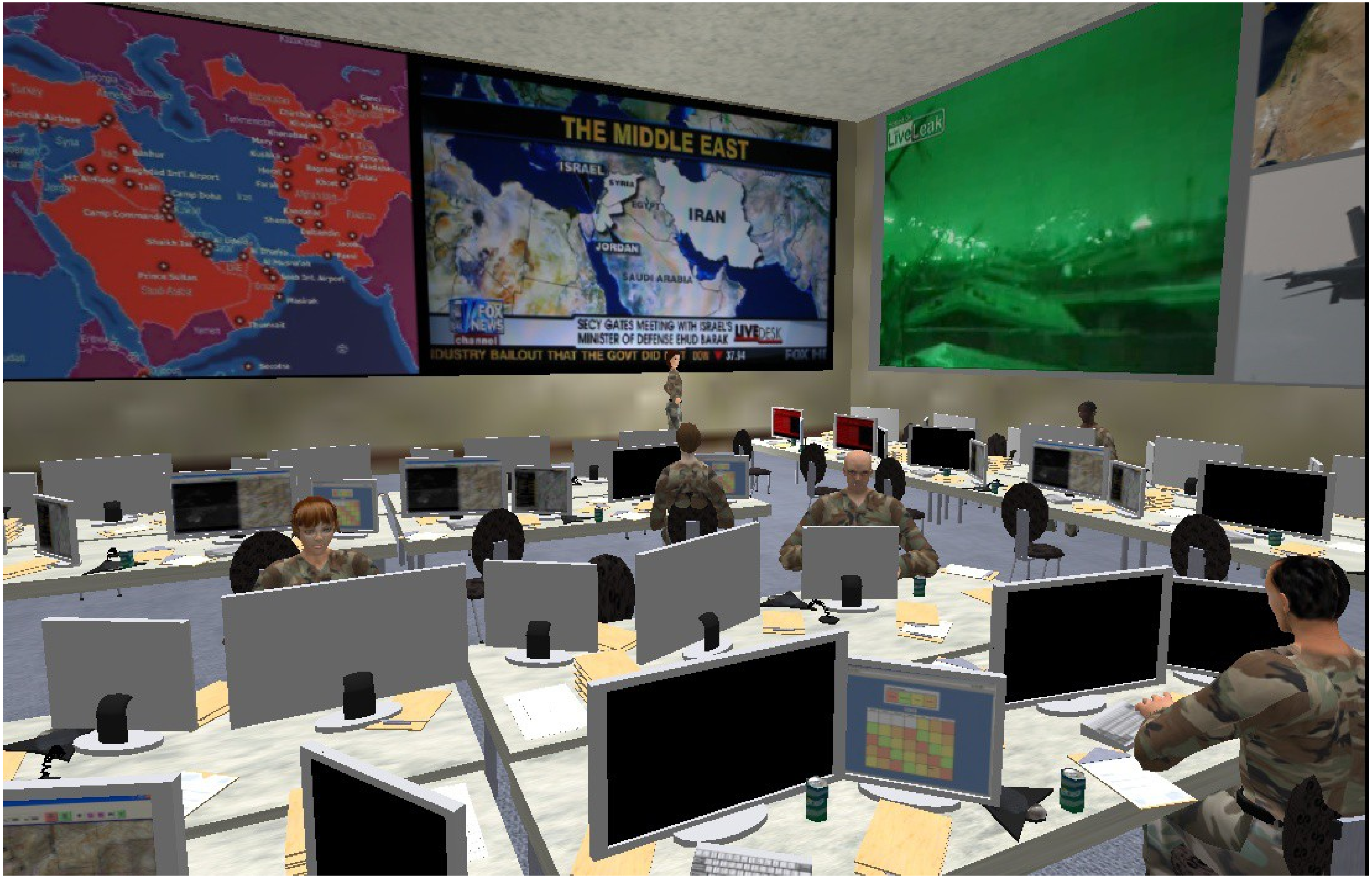
- Challenges of securing open architecture (OA) systems
- Specifying security requirements for software systems
- *Case study*: Securing the development and evolution of an OA C2 system within an agile, adaptive software ecosystem
- Discussion and conclusions

Challenges of securing open architecture (OA) C2 systems

Scacchi, W., Brown, C. and Nies, K. (2012). Understanding the Potential of Virtual Worlds for Decentralized Command and Control, *Proc. 17th. Intern. Command and Control Research and Technology Symposium* (ICCRTS), Paper-096, Fairfax, VA, June 2012.

Scacchi, W., Brown, C. and Nies, K. (2012). Understanding the Potential of Computer Games for Decentralized Command and Control, *Proc. 17th. Intern. Command and Control Research and Technology Symposium* (ICCRTS), Paper-104, Fairfax, VA, June 2012.

Virtual world for experimental studies in decentralized command and control centers using open source software components



Security challenges

- Security threats to software systems are increasingly multi-modal and distributed across system components.
- Physically isolated systems are vulnerable to external security attacks.
- What makes an OA C2 system secure changes over time, as new threats emerge and systems evolve.
- Need an approach *to continuously assure the security of evolving OA C2 systems* that is practical, scalable, robust, tractable, and adaptable.

Current security approaches

- Mandatory access control lists, firewalls;
- Multi-level security;
- Authentication (including certificate authority and passwords);
- Cryptographic support (including public key certificates);
- Encapsulation (including virtualization), hardware confinement (memory, storage, and external device isolation), and type enforcement capabilities;
- Secure programming practices;
- Data content or control signal flow logging/auditing;
- Honey-pots, traps, sink-holes;
- Security technical information guides for configuring the security parameters for applications and operating systems;
- Functionally equivalent but diverse multi-variant software executables.

Software systems/components *evolve*: what to do about security?

- Individual components evolve via revisions (e.g., security patches)
- Individual components are updated with functionally enhanced versions;
- Individual components are replaced by alternative components;
- Component interfaces evolve;
- System architecture and configurations evolve;
- System functional and security requirements evolve;
- System security policies, mechanisms, security components, and system configuration parameter settings also change over time.

Specifying the security requirements for OA software systems

Carefully specifying security policy obligations and rights

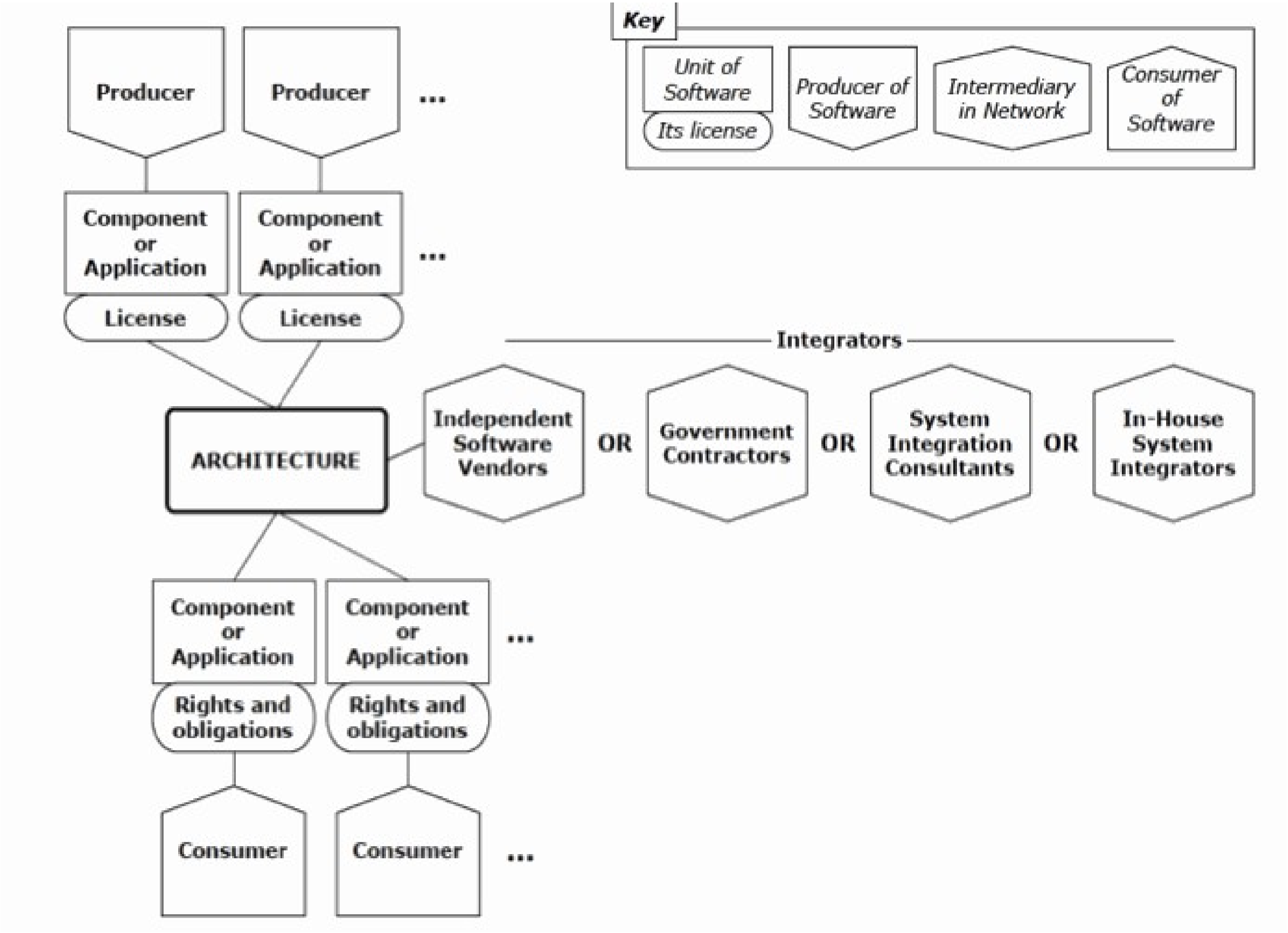
- The obligation for a user to verify his/her authority to see compartment T, by password or other specified authentication process
- The obligation for all components connected to specified component C to grant it the capability to read and update data in compartment T
- The obligation to reconfigure a system in response to detected threats, when given the right to select and include different component versions, or executable component variants.
- The right to read and update data in compartment T using the licensed component
- The right to add, update, replace specified component D in a specified configuration
- The right to add, update, or remove a security mechanism
- The right to update security policy L.

Case Study: Securing the development and evolution of an OA C2 system within an agile, adaptive software ecosystem

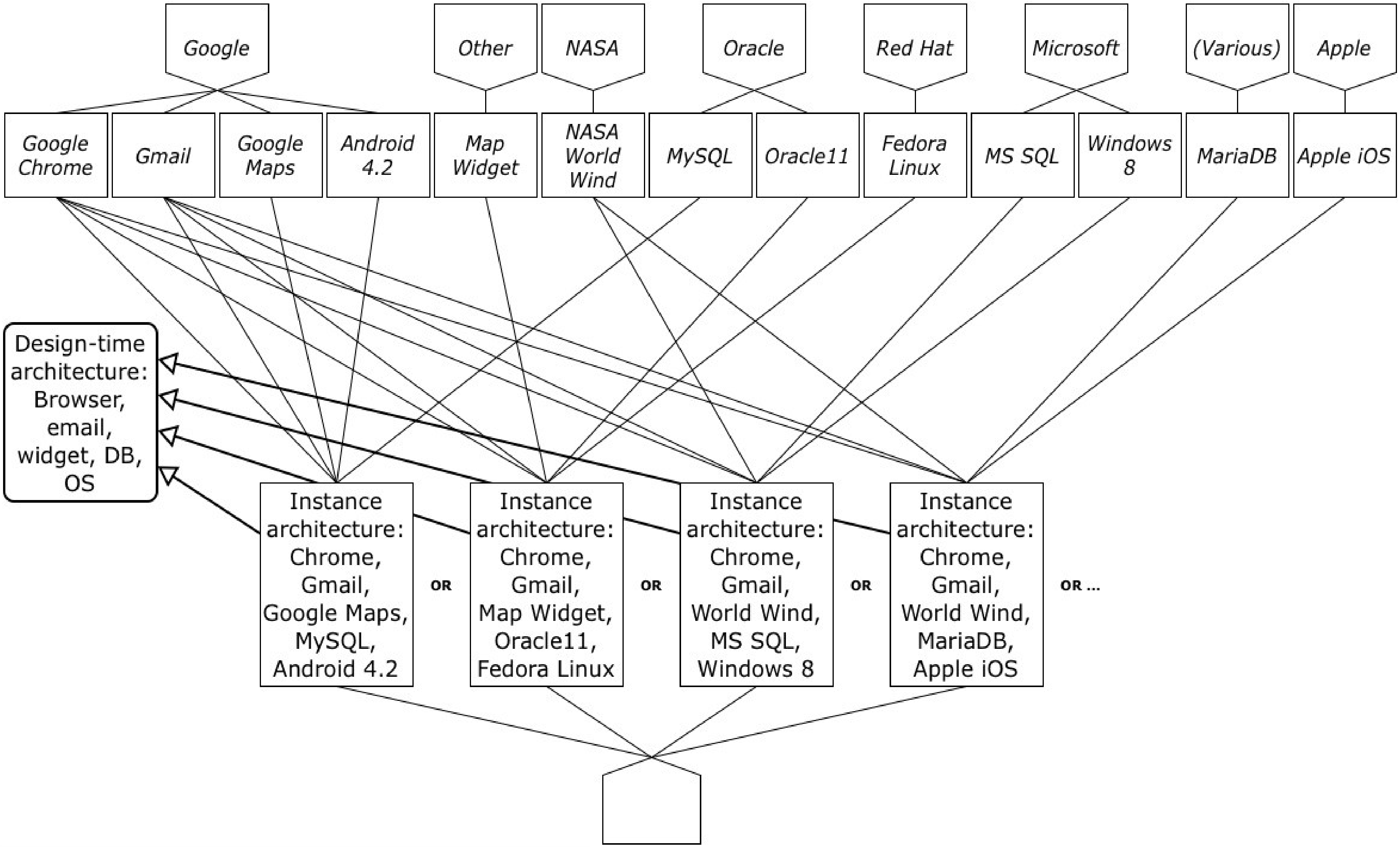
Software product lines?

- When functionally similar software components, connectors, or configurations exist,
- Such that equivalent alternatives, versions, or variants may be substituted for one another, then
- We have a strong relationship among these OA system elements that is called a *software product line*.
- Software product lines for OA systems enable support from agile, adaptive software (component) ecosystems
  - Reed, H., Benito, P., Collens, J. and Stein, F. (2012). Supporting Agile C2 with an Agile and Adaptive IT Ecosystem, *Proc. 17Th Intern. Command and Control Research and Technology Symposium* (ICCRTS), Paper-044, Fairfax, VA, June 2012.

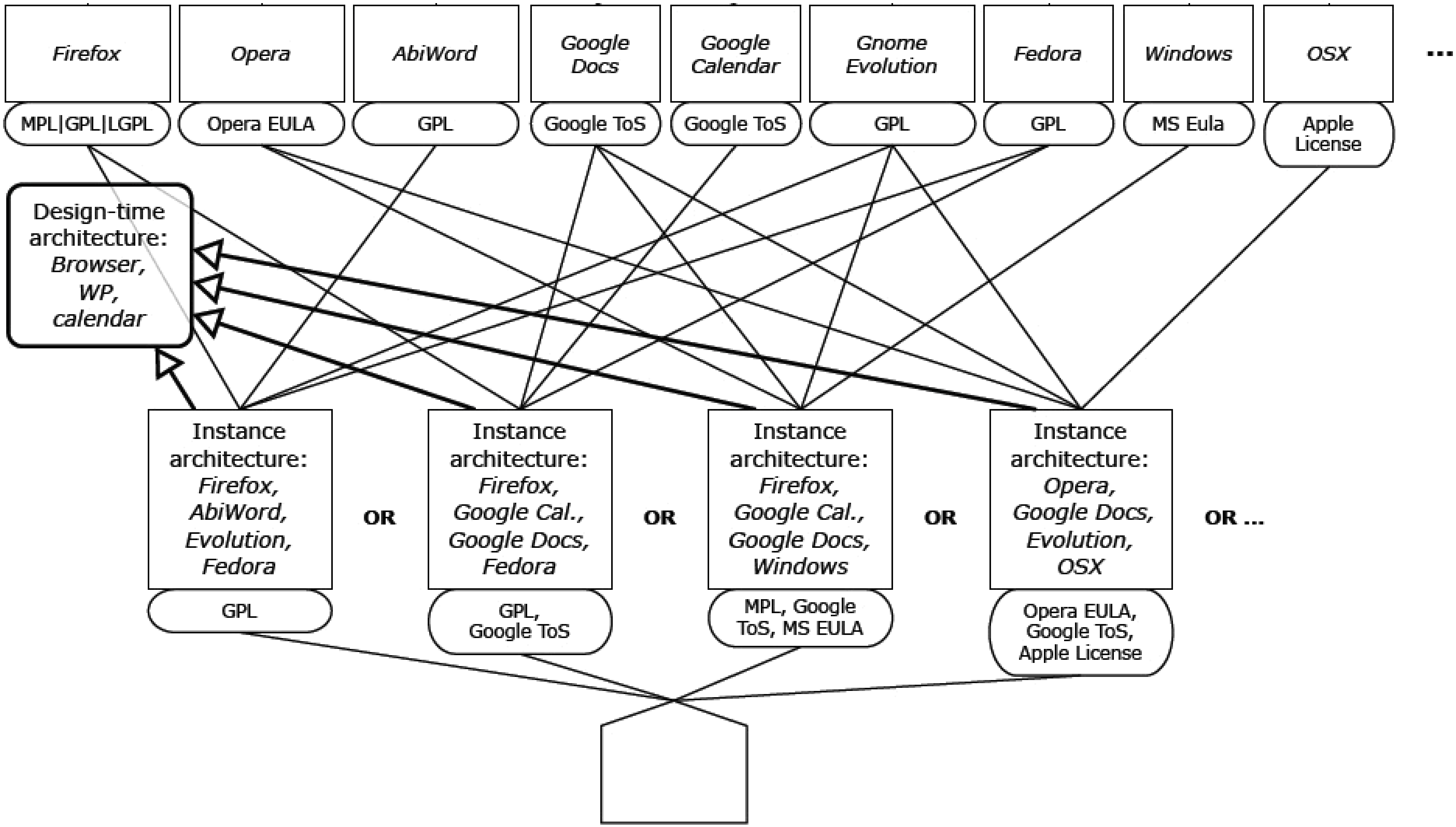
Software ecosystem of producers and the software components or application widgets for an enterprise system



Software ecosystem of components or application widgets for an OA system

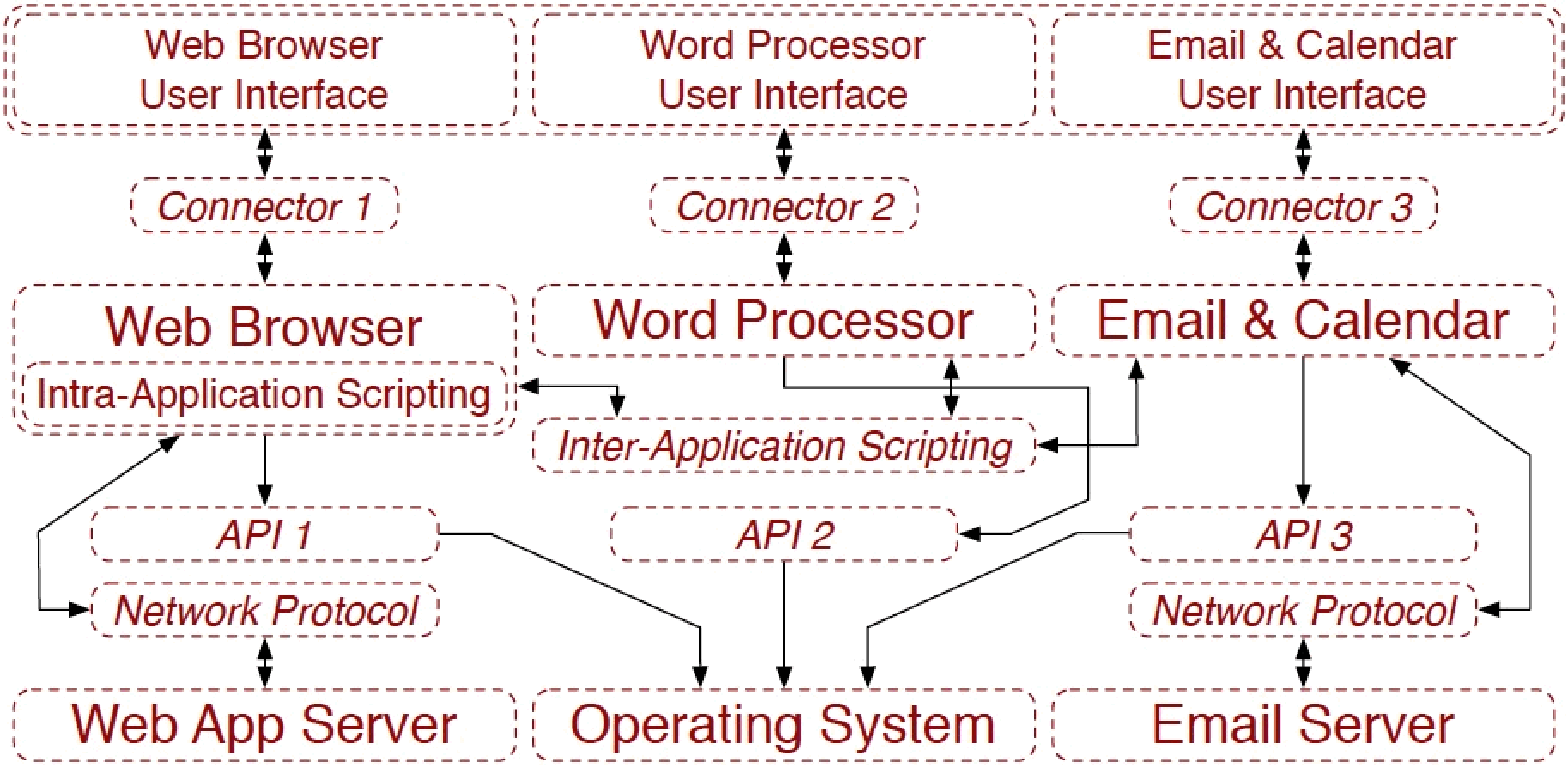

Google
Other
NASA
Oracle
Red Hat
Microsoft
(Various)
Apple
Google Chrome
Gmail
Google Maps
Android 4.2
Map Widget
NASA World Wind
MySQL
Oracle11
Fedora Linux
MS SQL
Windows 8
MariaDB
Apple iOS
Design-time architecture: Browser, email, widget, DB, OS
Instance architecture: Chrome, Gmail, Google Maps, MySQL, Android 4.2
OR
Instance architecture: Chrome, Gmail, Map Widget, Oracle11, Fedora Linux
OR
Instance architecture: Chrome, Gmail, World Wind, MS SQL, Windows 8
OR
Instance architecture: Chrome, Gmail, World Wind, MariaDB, Apple iOS
OR ...

Software product line that provides functionally similar components or applications compatible with an OA system design

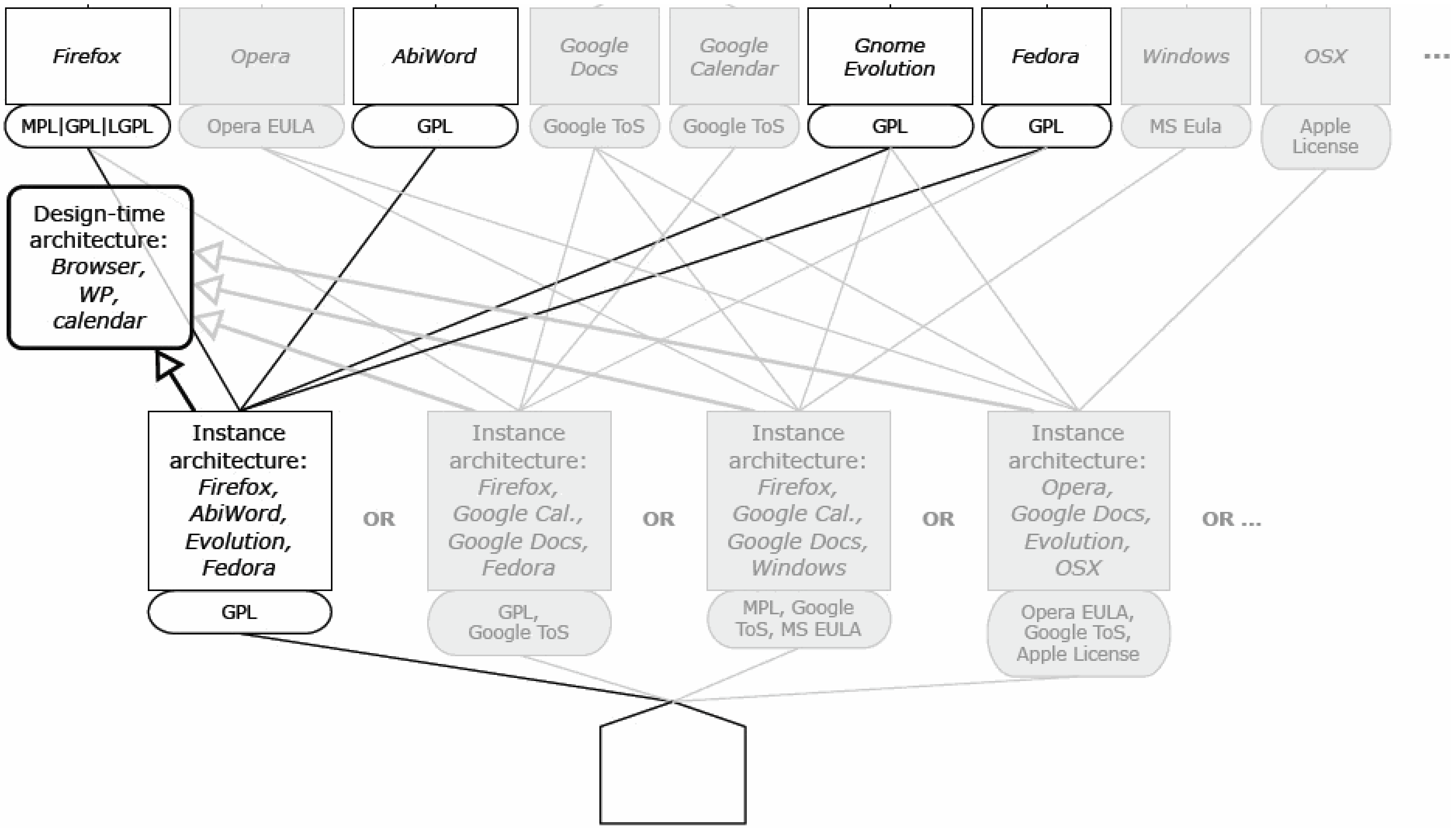

Firefox
MPL|GPL|LGPL
Opera
Opera EULA
AbiWord
GPL
Google Docs
Google ToS
Google Calendar
Google ToS
Gnome Evolution
GPL
Fedora
GPL
Windows
MS Eula
OSX
Apple License
...
Design-time architecture: Browser, WP, calendar
Instance architecture: Firefox, AbiWord, Evolution, Fedora
GPL
OR
Instance architecture: Firefox, Google Cal., Google Docs, Fedora
GPL, Google ToS
OR
Instance architecture: Firefox, Google Cal., Google Docs, Windows
MPL, Google ToS, MS EULA
OR
Instance architecture: Opera, Google Docs, Evolution, OSX
Opera EULA, Google ToS, Apple License
OR ...

ISR Institute for Software Research
University of California, Irvine

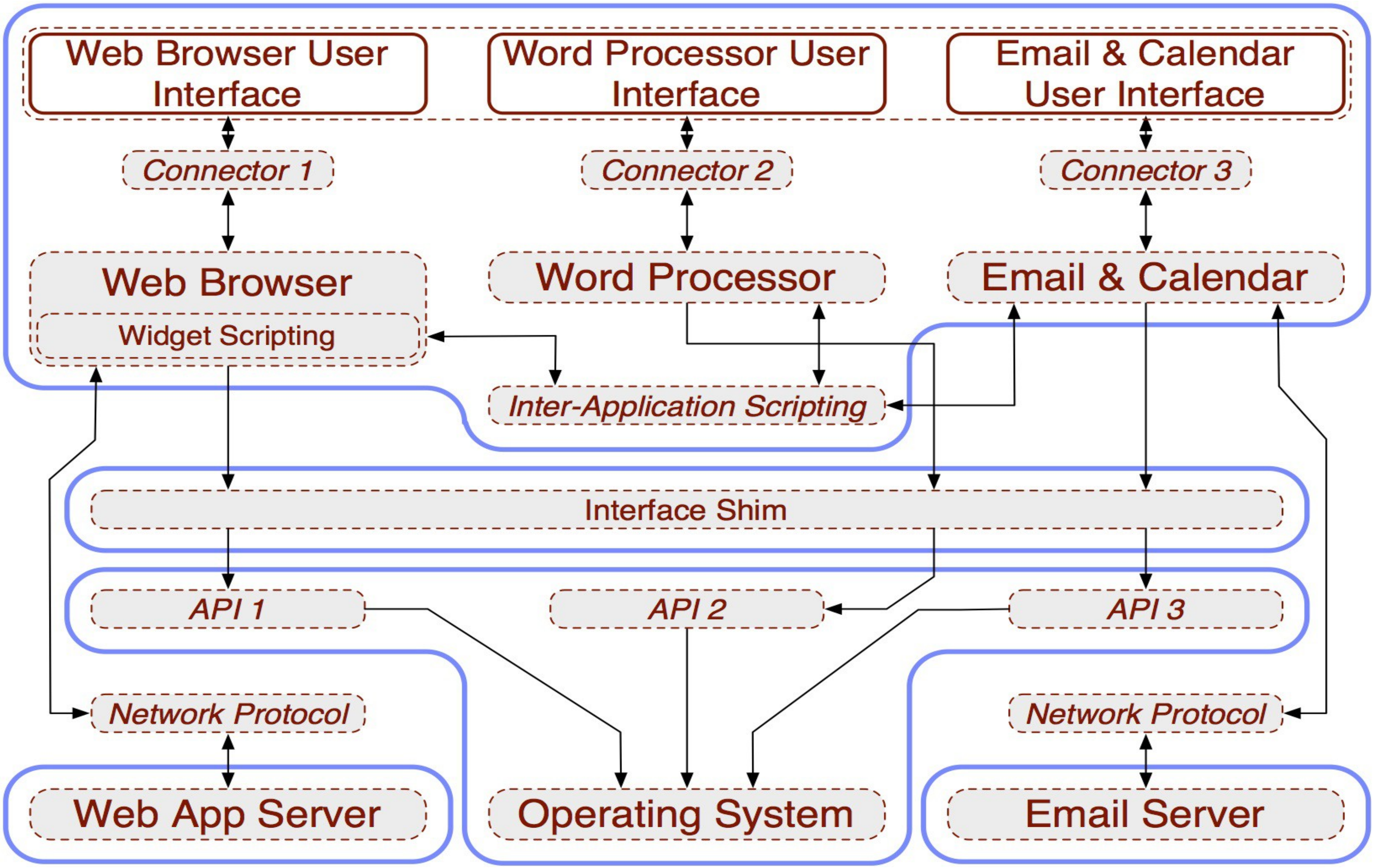
A *design-time* specification of an OA system that accommodates multiple alternative system configurations



A *build-time deployment selection* among alternative components that produce an integrated enterprise system within the product line

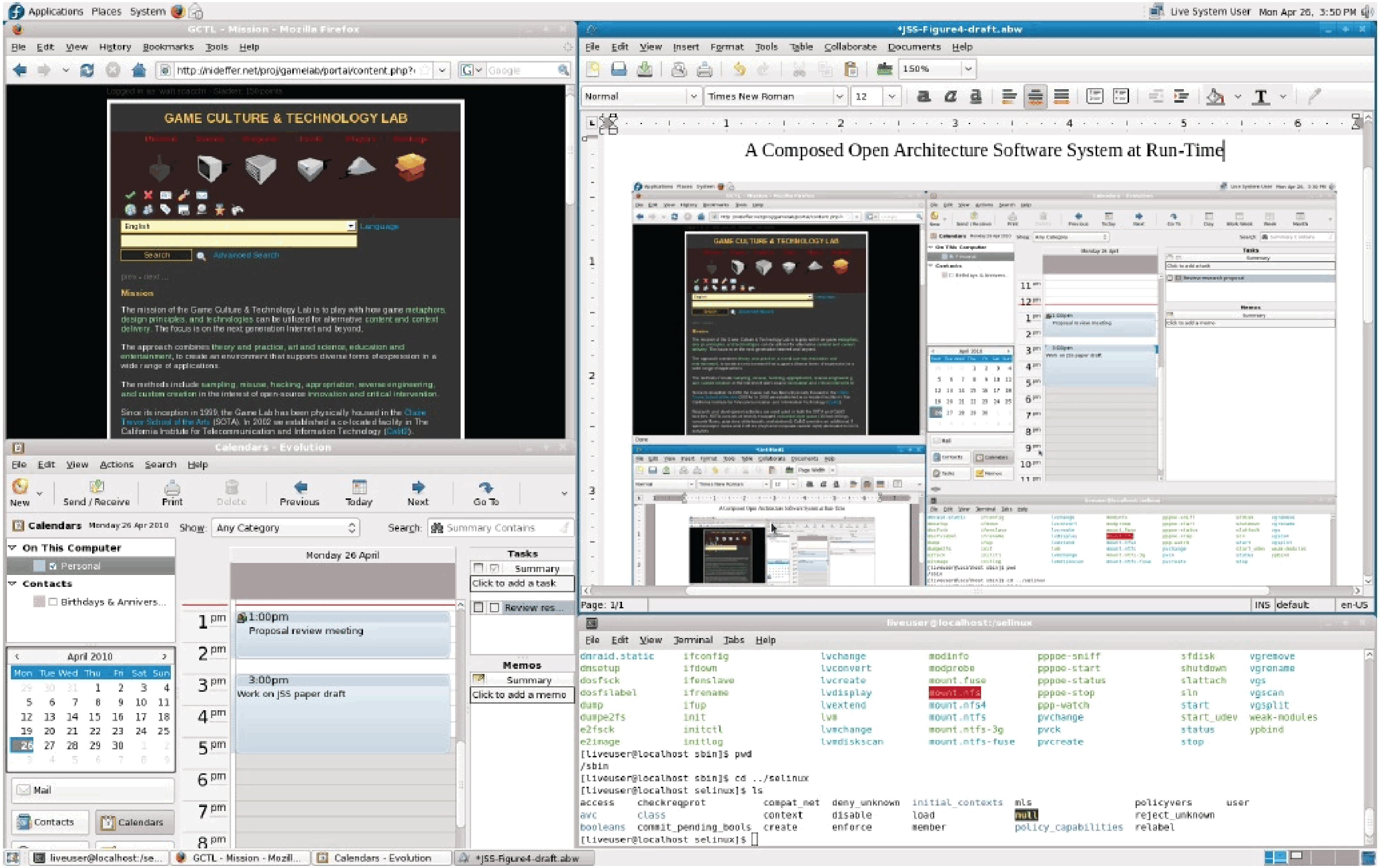

Firefox
MPL|GPL|LGPL
Opera
Opera EULA
AbiWord
GPL
Google Docs
Google ToS
Google Calendar
Google ToS
Gnome Evolution
GPL
Fedora
GPL
Windows
MS Eula
OSX
Apple License
...
Design-time architecture: Browser, WP, calendar
Instance architecture: Firefox, AbiWord, Evolution, Fedora
GPL
OR
Instance architecture: Firefox, Google Cal., Google Docs, Fedora
GPL, Google ToS
OR
Instance architecture: Firefox, Google Cal., Google Docs, Windows
MPL, Google ToS, MS EULA
OR
Instance architecture: Opera, Google Docs, Evolution, OSX
Opera EULA, Google ToS, Apple License
OR ...

A security capability specification encapsulating the *run-time deployment* configuration via multiple virtual machines



ISR Institute for Software Research
University of California, Irvine

An end-user *run-time deployment version* of selected components within enterprise system product line utilizing security library, **SELinux**, for enforcing mandatory obligations and rights.


Applications Places System
Live System User Mon Apr 26, 3:50 PM
GCTL - Mission - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://nideffer.net/proj/gamelab/portal/content.php?
Google
GAME CULTURE & TECHNOLOGY LAB
English
Language
Search
Advanced Search
prev - next ...
*JSS-Figure4-draft.abw
File Edit View Insert Format Tools Table Collaborate Documents Help
150%
Normal
Times New Roman
12
Page: 1/1
INS
default
en-US
Calendars - Evolution
File Edit View Actions Search Help
New
Send / Receive
Print
Delete
Previous
Today
Next
Go To
Calendars
Monday 26 Apr 2010
Show: Any Category
Search: Summary Contains
On This Computer
Personal
Contacts
Birthdays & Anniversaries...
April 2010
Mail
Contacts
Calendars
Tasks
Summary
Click to add a task
Review res...
Memos
Summary
Click to add a memo
liveuser@localhost:/selinux
File Edit View Terminal Tabs Help
liveuser@localhost:/se...
GCTL - Mission - Mozil...
Calendars - Evolution
*JSS-Figure4-draft.abw

Mission

The mission of the Game Culture & Technology Lab is to play with how game metaphors, design principles, and technologies can be utilized for alternative content and context delivery. The focus is on the next generation Internet and beyond.

The approach combines theory and practice, art and science, education and entertainment, to create an environment that supports diverse forms of expression in a wide range of applications.

The methods include sampling, misuse, hacking, appropriation, reverse engineering, and custom creation in the interest of open-source innovation and critical intervention.

Since its inception in 1999, the Game Lab has been physically housed in the Claire Trevor School of the Arts (SOTA). In 2002 we established a co-located facility in The California Institute for Telecommunication and Information Technology (Calit2).

A Composed Open Architecture Software System at Run-Time

Mon	Tue	Wed	Thu	Fri	Sat	Sun
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2
3	4	5	6	7	8	9

Monday 26 April

1 pm 1:00pm Proposal review meeting

2 pm

3 pm 3:00pm Work on JSS paper draft

4 pm

5 pm

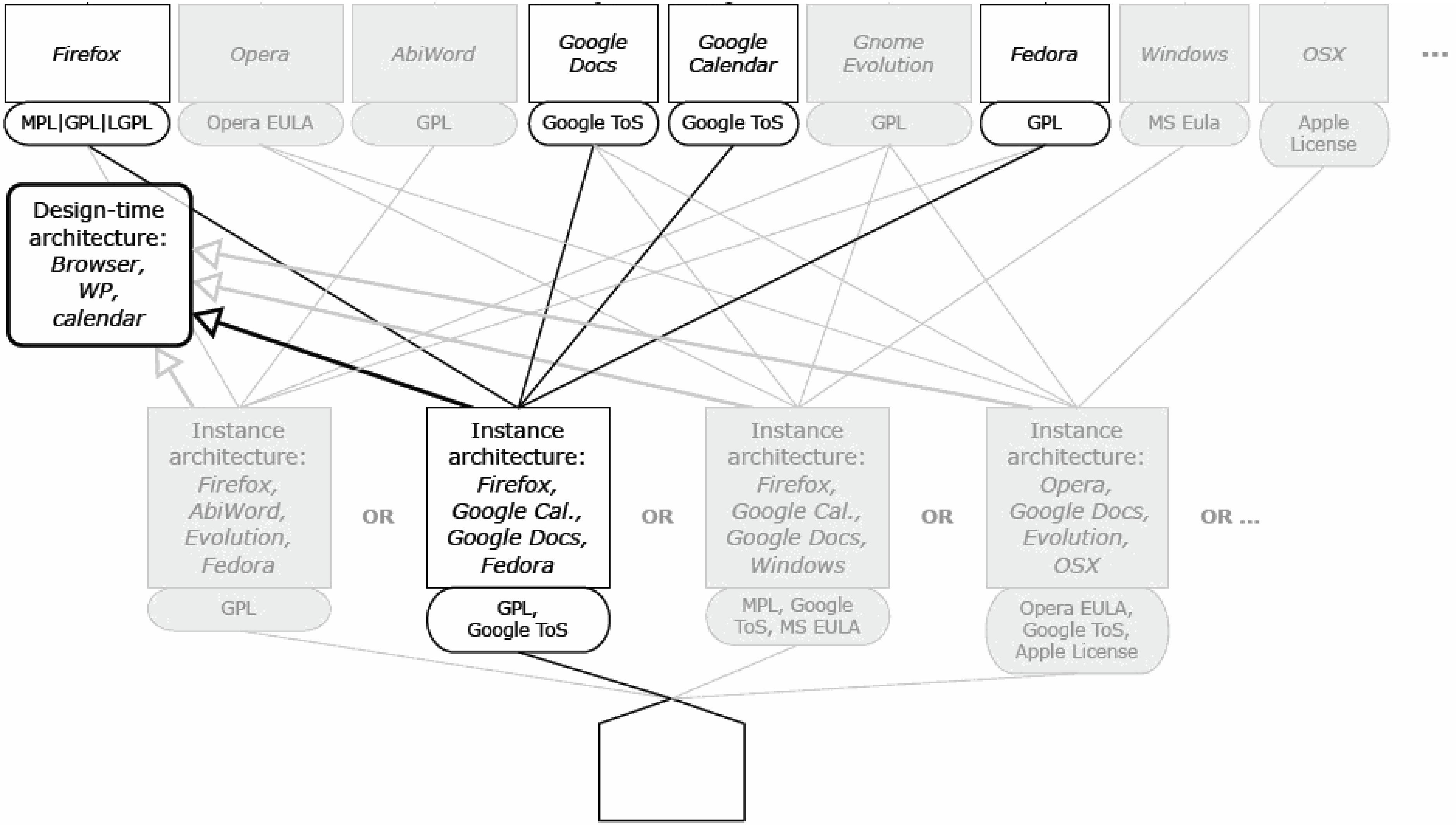
6 pm

7 pm

8 pm

```
dmraid.static     ifconfig          lvchange         modinfo           pppoe-sniff       sfdisk      vgremove
dmsetup           ifdown            lvconvert        modprobe          pppoe-start       shutdown    vgrename
dosfsck           ifenslave         lvcreate         mount.fuse        pppoe-status      slattach    vgs
dosfslabel        ifrename          lvdisplay        mount.nfs         pppoe-stop        sln         vgscan
dump              ifup              lvextend         mount.nfs4        ppp-watch         start       vgsplit
dumpe2fs          init              lvm              mount.ntfs        pvchange          start_udev  weak-modules
e2fsck            initctl           lvmchange        mount.ntfs-3g     pvck              status      ypbind
e2image           initlog           lvmdiskscan      mount.ntfs-fuse   pvcreate          stop
[liveuser@localhost sbin]$ pwd
/sbin
[liveuser@localhost sbin]$ cd ../selinux
[liveuser@localhost selinux]$ ls
access    checkreqprot          compat_net  deny_unknown  initial_contexts  mls                   policyvers      user
avc       class                 context     disable       load              null                  reject_unknown
booleans  commit_pending_bools  create      enforce       member            policy_capabilities   relabel
[liveuser@localhost selinux]$ 
```

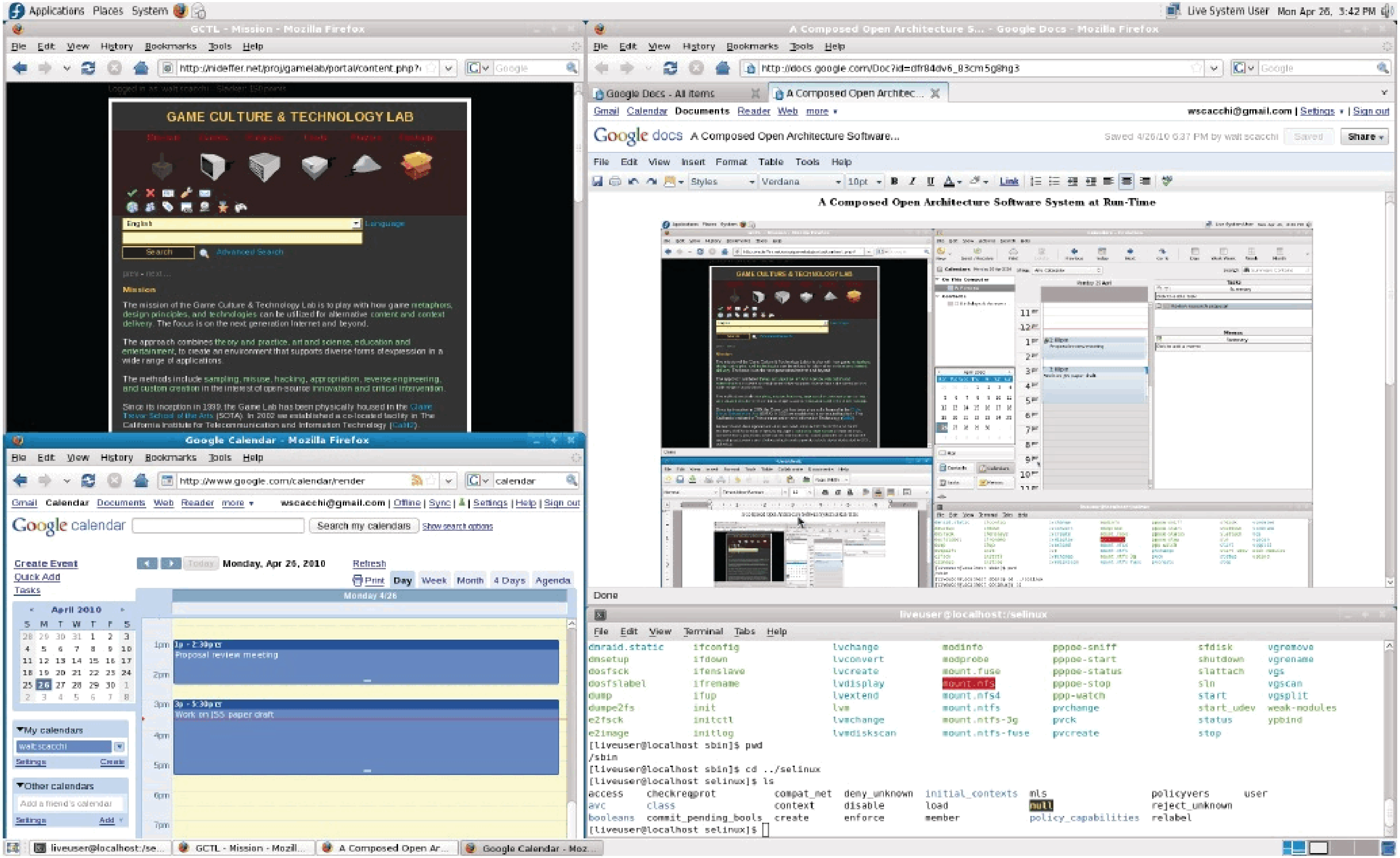
Adapting the *post-deployment system configuration*, using alternative but functionally similar components within the product line



Firefox
MPL|GPL|LGPL
Opera
Opera EULA
AbiWord
GPL
Google Docs
Google ToS
Google Calendar
Google ToS
Gnome Evolution
GPL
Fedora
GPL
Windows
MS Eula
OSX
Apple License
...
Design-time architecture: Browser, WP, calendar
Instance architecture: Firefox, AbiWord, Evolution, Fedora
GPL
OR
Instance architecture: Firefox, Google Cal., Google Docs, Fedora
GPL, Google ToS
OR
Instance architecture: Firefox, Google Cal., Google Docs, Windows
MPL, Google ToS, MS EULA
OR
Instance architecture: Opera, Google Docs, Evolution, OSX
Opera EULA, Google ToS, Apple License
OR ...

ISR Institute for Software Research
University of California, Irvine

An end-user view of the adapted alternative run-time system configuration



Applications Places System
Live System User Mon Apr 26, 3:42 PM
GCTL - Mission - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://nideffer.net/proj/gamelab/portal/content.php?
Google
A Composed Open Architecture S... - Google Docs - Mozilla Firefox
http://docs.google.com/Doc?id=dfr84dv6_83cm5g8hg3
Google Docs - All items
A Composed Open Architec...
Gmail Calendar Documents Reader Web more ▾
wscacchi@gmail.com | Settings ▾ | Sign out
Google docs
Saved 4/26/10 6:37 PM by walt scacchi
Saved
Share ▾
File Edit View Insert Format Table Tools Help
Styles Verdana 10pt B I U Link
Done
Google Calendar - Mozilla Firefox
http://www.google.com/calendar/render
calendar
Gmail Calendar Documents Web Reader more ▾
wscacchi@gmail.com | Offline | Sync | | Settings | Help | Sign out
Google calendar
Search my calendars
Show search options
Create Event
Quick Add
Tasks
Today
Refresh
Print Day Week Month 4 Days Agenda
My calendars
walt scacchi
Settings Create
Other calendars
Add a friend's calendar
Settings Add
liveuser@localhost:/selinux
File Edit View Terminal Tabs Help
liveuser@localhost:/se...
GCTL - Mission - Mozil...
A Composed Open Ar...
Google Calendar - Moz...

GAME CULTURE & TECHNOLOGY LAB

English

Language

Search

Advanced Search

prev - next ...

Mission

The mission of the Game Culture & Technology Lab is to play with how game metaphors, design principles, and technologies can be utilized for alternative content and context delivery. The focus is on the next generation Internet and beyond.

The approach combines theory and practice, art and science, education and entertainment, to create an environment that supports diverse forms of expression in a wide range of applications.

The methods include sampling, misuse, hacking, appropriation, reverse engineering, and custom creation in the interest of open-source innovation and critical intervention.

Since its inception in 1999, the Game Lab has been physically housed in the Claire Trevor School of the Arts (SCTA). In 2002 we established a co-located facility in The California Institute for Telecommunication and Information Technology (Calit2).

A Composed Open Architecture Software...

A Composed Open Architecture Software System at Run-Time

Monday, Apr 26, 2010

April 2010

S	M	T	W	T	F	S
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1
2	3	4	5	6	7	8

Monday 4/26

1pm 1p - 2:30p
Proposal review meeting

2pm

3pm 3p - 5:30p
Work on ISS paper draft

4pm

5pm

6pm

7pm

```
dmraid.static   ifconfig    lvchange     modinfo          pppoe-sniff   sfdisk      vgremove
dmsetup         ifdown      lvconvert    modprobe         pppoe-start   shutdown    vgrename
dosfsck         ifenslave   lvcreate     mount.fuse       pppoe-status  slattach    vgs
dosfslabel      ifrename    lvdisplay    mount.nfs        pppoe-stop    sln         vgscan
dump            ifup        lvextend     mount.nfs4       ppp-watch     start       vgsplit
dumpe2fs        init        lvm          mount.ntfs       pvchange      start_udev  weak-modules
e2fsck          initctl     lvmchange    mount.ntfs-3g    pvck          status      ypbind
e2image         initlog     lvmdiskscan  mount.ntfs-fuse  pvcreate      stop
[liveuser@localhost sbin]$ pwd
/sbin
[liveuser@localhost sbin]$ cd ../selinux
[liveuser@localhost selinux]$ ls
access    checkreqprot          compat_net  deny_unknown  initial_contexts  mls                   policyvers     user
avc       class                 context     disable       load              null                  reject_unknown
booleans  commit_pending_bools  create      enforce       member            policy_capabilities   relabel
[liveuser@localhost selinux]$ 
```

Discussion and conclusions

Discussion

- Our goal is to demonstrate a new approach to address challenges in the development and evolution of secure component-based OA C2 software systems.
- Future C2 systems require review and approval of security measures employed during the *design, implementation, deployment,* and *evolution* of OA systems.
- We seek to make this a simpler, more transparent, and more tractable process.

Conclusions (1)

- Our research demonstrates how complex OA systems can be designed, built, deployed, and evolved with alternative components within functionally similar system versions, to realize for overall system security.
- We described a scheme to specify and realize OA system configurations that are compatible with existing security mechanisms.
    - Our scheme does not assume that individual system elements must be secure before inclusion into the secured OA system's configuration.
- Central to our OA scheme is agile, adaptive software ecosystems and product lines integrated with security mechanisms.

Conclusions (2)

Next steps:

- Articulate the *process* how to simply and transparently specify and assess the security of OA C2 systems using streamlined security policy mechanisms.
- Develop and demonstrate a prototype *automated environment* that can support the modeling and analysis of OA system security policies and alternative version OA system configurations, in ways that address the diverse needs of software producers, system integrators and end-users.

Acknowledgements

Research described in this presentation was supported by grant #N00244-12-1-0067 from the Acquisition Research Program at the Naval Postgraduate School, and from grant #1256593 from the National Science Foundation.

No review, approval, or endorsement implied.